

# 3处理机调度

2018年10月22日 9:00



## ◆ 处理机调度的层次和调度算法的目标

1. 系统吞吐量、资源利用率、作业周转时间、响应及时性很大程度上取决于处理机调度性能好坏
2. 调度的实质是一种资源分配

### 一. 处理机调度的层次

#### 1. 高级调度(High Level Scheduling)

- 1) 作业调度: 调度对象是作业, 决定将哪些作业**从外存**的后备队列调入内存, 并创建进程、分配资源、放入就绪队列
- 2) 长程调度(LongTerm Scheduling): 周期长, 约几分钟, 一批作业完成才重新调, 运行频率低因而算法可以花较多时间
- 3) 出现在多道批处理系统

#### 2. 低级调度(Low Level Scheduling)

- 1) 进程调度: 调度对象是进程或内核级线程, 决定哪个进程分配到**处理机**
- 2) 短程调度(ShortTerm Scheduling): 运行频率高, 耗时极短, 算法不复杂
- 3) 是最基本的调度, 多道批处理、分时、实时系统都须配置这级调度

#### 3. 中级调度(Intermediate Level Scheduling)

- 1) 内存调度: 为提高内存利用率和系统吞吐量而挂起进程/改为就绪驻外存状态, 再由中级调度调入内存
- 2) 中程调度(Medium-Term Scheduling): 以上两者之间

- ☒ 3) 本质上就是**存储管理器对换**功能

### 二. 处理机调度算法的目标

#### 1. 各系统共同的目标:

- 1) 资源利用率: 让处理机等资源尽可能的保持忙碌状态  
(1)  $\text{cpu利用率} = \frac{\text{CPU有效工作时间}}{\text{CPU有效工作时间} + \text{CPU空闲等待时间}}$
- 2) 公平性: 诸进程都应获得合理的CPU时间, 不发生进程饥饿现象  
(1) 相对: 同类进程同服务, 不同紧急程度或重要性提供不同服务
- 3) 平衡性: 尽可能使各种CPU和外部设备处于忙碌状态
- 4) 策略强制执行: 包括安全策略, 需要时, 即使会延迟某些工作, 也要对所有制订的策略都准确执行

#### 2. 批处理系统特有的目标

- 1) 平均周转时间短: 尽可能提高系统资源利用率, 使大多数用户都满意  
(1) 作业周转时间: 作业被提交给系统开始, 完成为止的时间间隔  
(2) 包括: 外存后备队列上等、就绪队列上等, CPU上执行、阻塞时  
(3) ↑只有第一项只会发生一次  
(4) 平均周转时间&平均带权周转时间 ( $T_i$ 是周转时间,  $T_s$ 是系统提供

(处理机) 服务时间)

$$T = \frac{1}{n} \left[ \sum_{i=1}^n T_i \right] \quad W = \frac{1}{n} \left[ \sum_{i=1}^n \frac{T_i}{T_s} \right]$$

2) 系统吞吐量高: 尽可能单位时间内完成更多作业

(1) 尽可能选择短作业

3) 处理机利用率高, 这是衡量系统性能十分重要的指标

(1) 尽可能选择计算量大的作业

3. 分时系统特有的目标

1) 响应时间快: 分时系统调度算法重要准则

(1) **响应时间**: 从键盘提出请求到屏幕上显示出处理结果为止的时间

(2) 包括: 键入信息并传送进处理机、处理机处理信息、响应信息回送到终端显示器

2) 均衡性: 响应时间快慢与服务的复杂性相适应

4. 实时系统特有的目标

1) 截止时间保证: 尤其是HRT任务必须保证, 否则会有难以预料的后果, SRT也基本要保证

(1) 截止时间: 某任务开始执行/完成执行的最迟时间

2) 可预测性: 为了提高实时性, 需要预测

(1) 如多媒体系统最好用双缓冲, 提前处理下一帧

### 三. \*调度队列模型\*

1. 仅有进程调度的调度队列模型

1) 分时系统, 通常仅设置进程调度, 用户键入的命令和数据都直接送入内存。对于命令, 由 OS 为之建立一个进程

2) 系统可根据调度算法把处于就绪状态的进程组织成栈、树或无序链表

(1) 如, 分时系统中, 常把就绪进程组织成 FIFO 队列形式, 新进程被挂在就绪队列的末尾, 按时间片轮转方式运行

3) 每个进程在执行时都可能出现以下三种情况

(1) 任务在给定时间片内完成, 进程便在释放处理机后进入完成状态

(2) 任务在本次时间片内未完成, OS 便将该任务再放入就绪队列末尾

(3) 进程被阻塞, 被 OS 放入阻塞队列



图 3-1 仅具有进程调度的调度队列模型

2. 具有高级和低级调度的调度队列模型

1) 批处理系统, 不仅需要进程调度, 而且还需有作业调度。先由作业调度从外存的后备队列中选择一批作业调入内存, 并为它们建立进程, 送入就绪队列, 再由进程调度按照一定的进程调度算法选择一个进程, 把处理机分配给该进程

2) 批处理系统最常用的是最高优先权优先调度算法, 其就绪队列的形式:

(1) 优先权队列: 新进程根据优先权高低, 插入相应位置, 调度程序总把处理机分配给队首进程

(2) 无序链表：新进程总挂在链尾，每次调度时，依次比较该链中各进程的优先权，选出最高优先权。显然，效率较低

3) 设置多个阻塞队列：大、中型系统通常都按阻塞事件设置若干阻塞队列。系统较大时，阻塞队列中的进程数可以达到数百个，这将严重影响对阻塞队列操作的效率

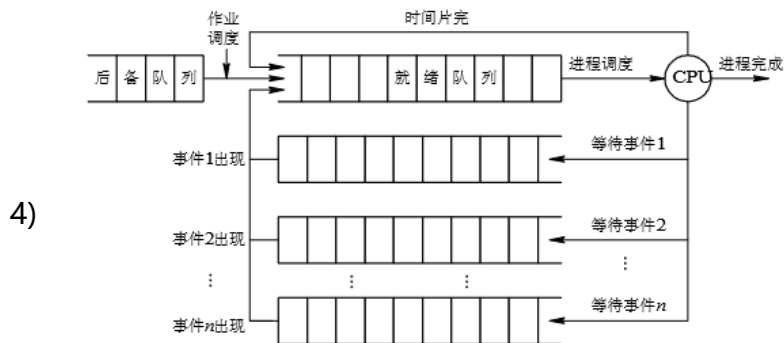


图 3-2 具有高、低两级调度的调度队列模型

### 3. 同时具有三级调度的调度队列模型

1) 当在 OS 中引入中级调度后，人们可把进程的就绪状态分为内存就绪(表示进程在内存中就绪)和外存就绪(进程在外存中就绪)。类似地，也可把阻塞状态进一步分成内存阻塞和外存阻塞两种状态。在调出操作的作用下，可使进程状态由内存就绪转为外存就绪，由内存阻塞转为外存阻塞；在中级调度作用下，又可使外存就绪转为内存就绪

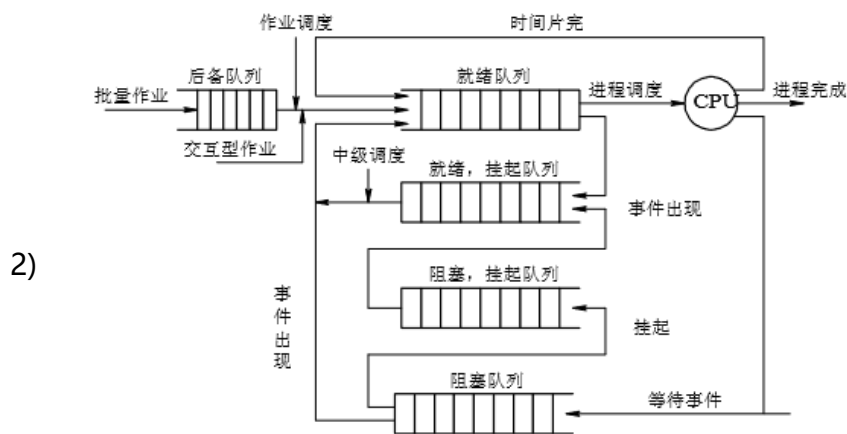


图 3-3 具有三级调度时的调度队列模型

- i.
- ii.
- iii.
- iv.
- v.
- vi.
- vii. -----我是底线-----